

SQL Joins

Bases de Datos Relacionales

Temas

- Cross Join
- Join
- Alias para Tabla
- Self Join

Cross Join o Producto Catersiano

SELECT *
FROM citizen, postcode

citizen

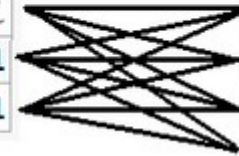
ID	NAME	POSTAL	LEADER
1	Smith	2200	<i>null</i>
2	Green	31006	1
3	Jensen	623	1

postcode

ID	POSTAL	CITY	AREA
1	2200	BigCity	Geancy
2	31006	SmallTown	Snizkim
3	31006	Settlement	Moon
4	78567390	LookoutTowerX89	Space

citizen

ID	NAME	POSTAL	LEADER
1	Smith	2200	<i>null</i>
2	Green	31006	1
3	Jensen	623	1



postcode

ID	POSTAL	CITY	AREA
1	2200	BigCity	Geancy
2	31006	SmallTown	Snizkim
3	31006	Settlement	Moon
4	78567390	LookoutTowerX89	Space

ID	NAME	POSTAL	LEADER	ID	POSTAL	CITY	AREA
1	Smith	2200	<i>null</i>	1	2200	BigCity	Geancy
1	Smith	2200	<i>null</i>	2	31006	SmallTown	Snizkim
1	Smith	2200	<i>null</i>	3	31006	Settlement	Moon
1	Smith	2200	<i>null</i>	4	78567390	LookoutTowerX89	Space
2	Green	31006	1	1	2200	BigCity	Geancy
2	Green	31006	1	2	31006	SmallTown	Snizkim
2	Green	31006	1	3	31006	Settlement	Moon
2	Green	31006	1	4	78567390	LookoutTowerX89	Space
3	Jensen	623	1	1	2200	BigCity	Geancy
3	Jensen	623	1	2	31006	SmallTown	Snizkim
3	Jensen	623	1	3	31006	Settlement	Moon
3	Jensen	623	1	4	78567390	LookoutTowerX89	Space

Join


Trae los registros que están en ambas tablas haciendo 'join' por columnas que tienen el mismo valor en ambas tablas

```
SELECT *  
FROM citizen,postalcode  
WHERE citizen.postal =  
postalcode.postal
```

ID	NAME	POSTAL	LEADER
1	Smith	2200	<i>null</i>
2	Green	31006	1
3	Jensen	623	1

ID	POSTAL	CITY	AREA
1	2200	BigCity	Geancy
2	31006	SmallTown	Snizkim
3	31006	Settlement	Moon
4	78567390	LookoutTowerX89	Space

ID	NAME	POSTAL	LEADER
1	Smith	2200	<i>null</i>
2	Green	31006	1
3	Jensen	623	1



ID	POSTAL	CITY	AREA
1	2200	BigCity	Geancy
2	31006	SmallTown	Snizkim
3	31006	Settlement	Moon
4	78567390	LookoutTowerX89	Space

ID	NAME	POSTAL	LEADER	ID	POSTAL	CITY	AREA
1	Smith	2200	<i>null</i>	1	2200	BigCity	Geancy
2	Green	31006	1	2	31006	SmallTown	Snizkim
2	Green	31006	1	3	31006	Settlement	Moon

Join

Liste el nombre del 'citizen' y la ciudad donde vive.

ID	NAME	POSTAL	LEADER
1	Smith	2200	<i>null</i>
2	Green	31006	1
3	Jensen	623	1

ID	POSTAL	CITY	AREA
1	2200	BigCity	Geancy
2	31006	SmallTown	Snizkim
3	31006	Settlement	Moon
4	78567390	LookoutTowerX89	Space

```
SELECT citizen.name, postalcode.city  
FROM citizen , postalcode  
WHERE citizen.postal =  
postalcode.postal  
AND citizen.leader is not null
```

ID	NAME	POSTAL	LEADER
1	Smith	2200	<i>null</i>
2	Green	31006	1
3	Jensen	623	1

ID	POSTAL	CITY	AREA
1	2200	BigCity	Geancy
2	31006	SmallTown	Snizkim
3	31006	Settlement	Moon
4	78567390	LookoutTowerX89	Space

NAME	CITY
Smith	BigCity
Green	SmallTown
Green	Settlement

Join usando llaves PK y FK

- Observe el uso de llaves naturales y no de llaves sustitutas para la PK

The diagram illustrates a join between two tables: **Product** and **Company**. The **Product** table has columns PName, Price, Category, and Manufacturer. The **Company** table has columns CName, StockPrice, and Country. An orange arrow points from the CName column in the **Company** table to the Manufacturer column in the **Product** table, indicating a foreign key relationship. Red arrows point to the PName and CName columns, labeling them as 'Key'. A red arrow points to the Manufacturer column, labeling it as 'Foreign key'.

<u>PName</u>	Price	Category	Manufacturer
Gizmo	\$19.99	Gadgets	GizmoWorks
Powergizmo	\$29.99	Gadgets	GizmoWorks
SingleTouch	\$149.99	Photography	Canon
MultiTouch	\$203.99	Household	Hitachi

<u>CName</u>	StockPrice	Country
GizmoWorks	25	USA
Canon	65	Japan
Hitachi	15	Japan

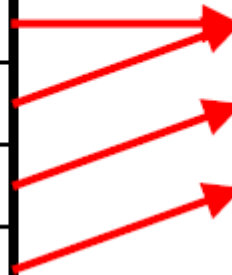
Join usando llaves PK y FK

Product

<u>PName</u>	Price	Category	Manufacturer
Gizmo	\$19.99	Gadgets	GizmoWorks
Powergizmo	\$29.99	Gadgets	GizmoWorks
SingleTouch	\$149.99	Photography	Canon
MultiTouch	\$203.99	Household	Hitachi

Company

<u>CName</u>	StockPrice	Country
GizmoWorks	25	USA
Canon	65	Japan
Hitachi	15	Japan



Product(pname, price, category, manufacturer(FK1)) Product.manufacturer is FK a Company

Company(cname, stockprice, country)

Join

- Encuentre los nombres y precios de todos los productos de menos de \$ 200 fabricados en Japón

Product

PName	Price	Category	Manufacturer
Gizmo	\$19.99	Gadgets	GizmoWorks
Powergizmo	\$29.99	Gadgets	GizmoWorks
SingleTouch	\$149.99	Photography	Canon
MultiTouch	\$203.99	Household	Hitachi

Company

CName	StockPrice	Country
GizmoWorks	25	USA
Canon	65	Japan
Hitachi	15	Japan

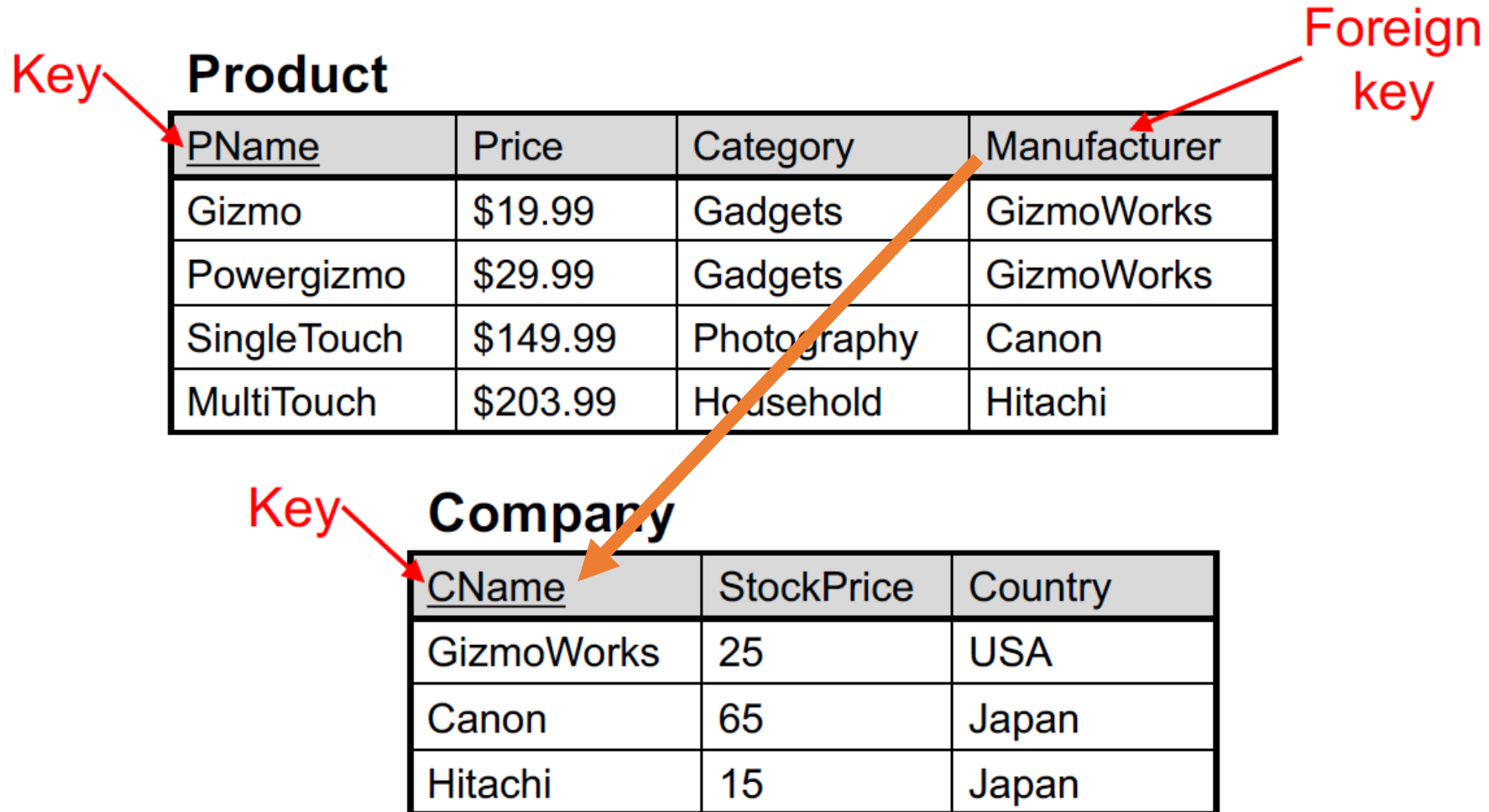
PName	Price
SingleTouch	\$149.99

Join

- Encuentre los nombres y precios de todos los productos de menos de \$ 200 fabricados en Japón

```
SELECT Product.pName, Product.Price
FROM   Product, Company
WHERE Product.manufacturer = Company.cName
AND    Company.country = 'Japan'
AND    Product.Price <= 200
```

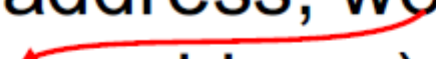
Taller 1



Alias para Tabla


- La columna 'address' existe en ambas tablas, esto es ambigüedad

Person (pName, address, works_for)
University (uName, address)



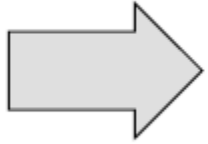
Error

```
SELECT DISTINCT pName, address  
FROM Person, University  
WHERE works_for = uName
```



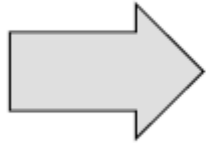
Alias para Tabla

- La columna 'address' existe en ambas tablas, esto es ambigüedad



```
SELECT DISTINCT Person.pName, University.address  
FROM      Person, University  
WHERE     Person.works_for = University.uName
```

Alias para Tabla

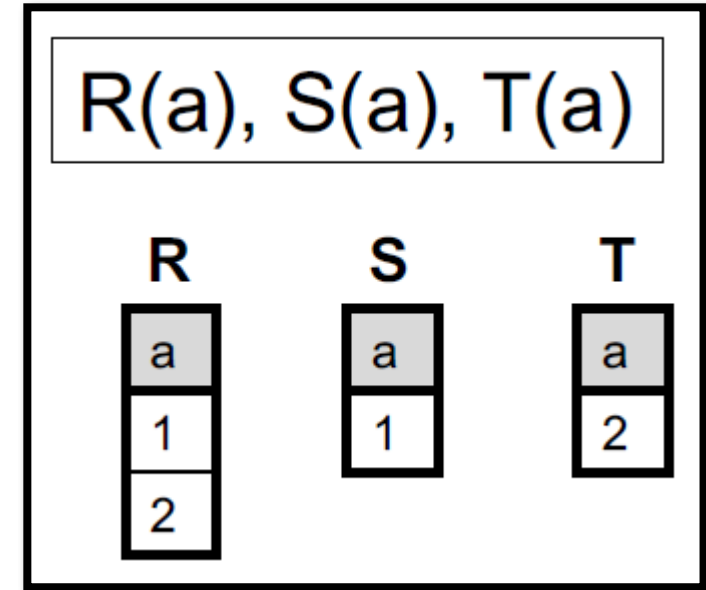


```
SELECT DISTINCT X.pName, Y.address  
FROM    Person as X, University Y  
WHERE   X.works_for = Y.uName
```

- X y Y son alias
- Se puede usar 'as X' o 'as Y'

Joins con más de 2 tablas

```
SELECT DISTINCT R.a, S.a, T.a  
FROM R, S, T  
WHERE R.a=S.a  
AND R.a=T.a
```



Self Join

- Encuentre los nombres de los productos cuyo precio es mayor al precio del producto llamado '**Powergizmo**'
 - Usted podría escribir

SELECT pName

FROM Product

Where Price > 29.99

¿Pero cuando el precio cambie?

No serviría la consulta

Usted debe usar la información conocida '**Powergizmo**'

Product			
<u>PName</u>	Price	Category	Manufacturer
Gizmo	\$19.99	Gadgets	GizmoWorks
Powergizmo	\$29.99	Gadgets	GizmoWorks
SingleTouch	\$149.99	Photography	Canon
MultiTouch	\$203.99	Household	Hitachi

Self Join

Product			
<u>PName</u>	Price	Category	Manufacturer
Gizmo	\$19.99	Gadgets	GizmoWorks
Powergizmo	\$29.99	Gadgets	GizmoWorks
SingleTouch	\$149.99	Photography	Canon
MultiTouch	\$203.99	Household	Hitachi

```
SELECT DISTINCT pName
FROM Product as P1, Product as P2
where P1.pName='Powergizmo'
And P2.pName <> 'Powergizmo'
and P2.price > P1.price
```

pName
SingleTouch
MultiTouch